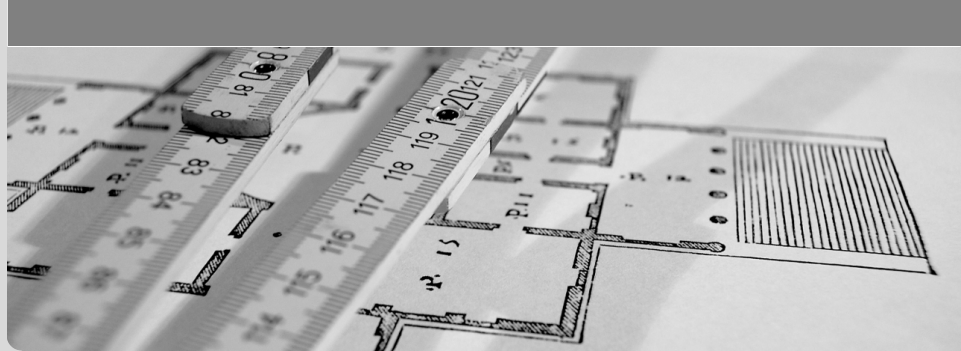


SMW System Testing with Selenium

Benedikt Kämpgen | 18th of September 2010



1 Introduction

- Testing, who needs that?
- Selenium – so, what?
- Think big, start small.

2 SMW System Testing with Selenium

- What requirements?
- What tests?
- How to use Selenium Framework?
- How to create Selenium tests?

3 Conclusion

- Lessons Learned
- More comin'?
- Discussion & Questions

Testing, who needs that?



- *[Wikimania 2010] – Important topics:*
 - stability
 - extensibility
 - code quality
- *[SMW-devel 2010] – Code quality and how to improve it:*
 - (1) Create and run Selenium tests.
 - (2) Create and run PHPUnit tests.
 - (3) Review existing/new code and extend documentation

Selenium – so, what?



- *Unit Testing*: Bottom-Up –
test of units = smallest testable part of an application, e.g., Classes/Methods. (PHPUnit)
- *System Testing*: Top-Down –
test of requirements = usage of the system in black box manner (no understanding of design/logic)

- Why Selenium?
 - Open Source System Testing Tool.
 - Remote control of browser(s) on a Selenium server to test a web-based application.
 - Tests in PHP (also Java, Ruby, HTML).
 - Wikimedia Foundation Project [1] deploying Selenium (code quality, infrastructure).
- Possible: Everything, that JavaScript can do – insert, click, press key, handle cookies, drag&drop. . .
- Not possible: Things, JavaScript cannot do – switch tabs, control message boxes, change browser settings.

[1] <http://www.mediawiki.org/wiki/Selenium/Deployment>

■ BIG

- Selenium Grid by the Foundation with Selenium Server and various browsers
- Automatic Test Environments (e.g. installation, configuration, database)
- Automatic MW/SMW/extensions revision testing with report to code review (continuous integration)

■ SMALL

- Selenium Server and MW/SMW/extension application and Selenium tests on same machine
- Regression tests for SMW core and extensions (development, combinations, browsers)

What requirements? (1)

User Manual => Requirements => Tests [1]


- Properties of various types can be created.
- Subcategories can be created.
- Factbox shows properties and values of a page.
- Silent Annotations can be used to have string property values with square brackets.
- Based on another property, a recurring event can be created.
- Inline Query shows certain pages and their properties.
- A query result can be sorted by clicking on a column.
- Inline Queries can show single item of properties of type record.

[1] http://www.semantic-mediawiki.org/w/index.php?title=SMW_System_Testing_with_Selenium

What requirements? (2)

- Semantic Search can run same queries as in Inline Queries.
- A custom unit with automatic conversion can be created and used for annotating a page.
- Synonyms of Properties can be created.
- ...

Set up? (1)

- 
- Create subcategory of "Actor" "Layman_Actor"
 - Create property "height", "located in", "birthdate"
 - Create property "lived in" of type record with has fields page (location) and date (when)
 - Create property "keyword" of type string
 - Create property "birthday" of type date
 - Create property type "Type:Power" that "corresponds to" "1 Watt" and "0.0013410220 Horsepower"
 - Create property "strength" of type "Power"

Set up? (2)

- Create page "Sean Connery" of category "Actor" with "born in" "England", with "height" "180 cm", with "birthdate" "15.1.1955", with "lived in" "Scotland; 1990".
- Annotate "Sean Connery" with "keyword" "[[strong]]" using silent annotation
- Annotate "Sean Connery" with yearly recurring event of "birthdate"
- Annotate "Sean Connery" with "has power" "200 Horsepower"
- Create page "Pinocchio" of category "Layman_Actor" with "born in" "Italy", with "height" "50 cm" with "birthdate" "15.1.1930", with "lived in" "Milan;1940"

Set up? (3)

- Create property "surface area" of type "number"
- Create Template "surface area" that displays the surface area and annotates the page
- Create page "England" with "located in" "Europe" using an alternate text annotation
- Annotate "England" with "surface area" "25e4".
- Create synonym of "located in" "lies in" by a redirect

Run tests? (1)



- Visit “Sean Connery”, retrieve and check values in factbox, then visit Special:Browse and type in Sean Connery, and check again.
 - Checking whether Factbox exists, contains right strength, right keyword, property birthday.
 - Checking whether Special:Browse contains keyword [[strong]]

Run tests? (2)

- Create Inline Query asking for “Actors” (and therefore “Layman_Actors”) and “born in” and “height” and “lived in#where |+index=1”, and “birthday” and asking for yes/no of the category “Layman_Actor”; sorted by descending height.
 - Assert whether result table exists and shows the wanted.
 - Checking whether Special:Browse contains keyword strong
 - Click on column for sort by ascending height and check, again.
 - Check, whether record property item is shown correctly.
- Visit Special:Ask and create same query, using auto-completion, and check again.
- Ask for Actors "born in" that is “lies in” “Europe” and has “surface area” bigger than “20e4”, and check whether “Sean Connery” is shown.

Tear down?



- Purge created wiki content so that other tests are not influenced.
- E.g. empty pages.

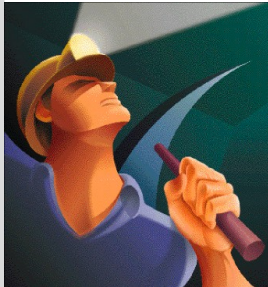
How to use Selenium Framework?

- Download and start selenium-server.jar [1].
- Download trunk version of MediaWiki and Semantic MediaWiki.
- Set up MW Selenium Framework [2] (due to changes!): URL, Login, Browser, TestSuites.
- Run Selenium tests from console.
- Demo...

[1] <http://seleniumhq.org/projects/remote-control>

[2] <http://www.mediawiki.org/wiki/SeleniumFramework>

How to create Selenium tests?



- Location: EXTENSION/tests/selenium
- Structure: Test Fixture -> Test Suite -> Test Case -> setUp()/runTests()/tearDown();
- Some code: Demo...
 - Create record property. Check record output from Inline Query.
 - Sort Inline Query Result.
 - Autocompletion in Special:Ask.


■ Robustness:

- How to find elements to test? Firebug, DOM Inspector, XPather
- Sensible to changes. MediaWiki: 1.15.4 -> 1.16.0;
Login: wpLoginattempt -> wpLoginAttempt
- More sensible to changes without name/ID, e.g.,
Semantic Search Page Input Fields.

■ Effort/Performance:

- 5% – 10% of SMW tested (2 days, 34 seconds).
- Writing tests not too difficult.
- Running time of tests could be an issue.



- 
- Selenium Framework still at the beginning
 - Framework for common tests (e.g., login).
 - Best practices for organizing tests (fast, modular, robust, easy-to-maintain).
 - Best practices for visualizing tests (hints to *possible* errors).
 - What we'd like to do
 - Improve on SMW Selenium tests (and use them!).
 - Stay up-to-date about WMF Selenium Framework.
 - Help developers to write/use SMW extension Selenium tests.

- Feedback...? Questions...?
- Anyone having system testing experiences (s)he wants to share?
- Tests for having less experienced developers get involved more quickly?
- Refactoring of the user manual to have tests properly related to it?
- Thank you!

